

*Revista científica CENTROS*15 de julio de 2019 – Vol. 8 No. 2 ISSN: **2304-604X***pp.120-134**Recibido:06/03/19 ; Aceptado:3/05/19*

Se autoriza la reproducción total o parcial de este artículo, siempre y cuando se cite la fuente completa y su dirección electrónica.

<https://revistas.up.ac.pa/index.php/centros>*indexada en*<http://www.latindex.unam.mx/><http://miar.ub.edu/issn/2304->

## Calidad del Software una Perspectiva Continua

### Software Quality a Continuous Perspective

Ing. Aizprua, Saulo; Ing. Ortega Amable<sup>1</sup>; Ing. Von Chong, Lilly

Universidad de Panamá.

saulo.aizprua@up.ac.pa, amable.ortega@up.ac.pa, lchong@hotmail.es

## RESUMEN

Este documento presenta una definición de calidad de software contextualizada, es decir según el área puede verse desde la perspectiva del producto, proceso o equipo de software. A esta definición le acompañan definiciones de características que no pueden hacer falta, como los son: la funcionalidad, confiabilidad, usabilidad, eficiencia, facilidad de mantenimiento y portabilidad de producto de software. Es al ver la calidad del software desde estas perspectivas que se valora la comprensión de estándares que faciliten métricas para una evaluación más realista dependiendo de lo que se esté evaluando, ya sea el software como producto, el equipo de desarrollo o el proceso de desarrollo, por tal razón se describen estándares como: ISO 12207; para evaluar ciclos de vida del software, ISO 9001; para asegurar calidad en el diseño y desarrollo, ISO 9002; Sistemas de calidad- Modelo de aseguramiento de la calidad en la producción e instalación, ISO 9126; Calidad del producto, ISO 14598; Evaluación de productos de software. Así

como contamos con estándares para atender estas perspectivas de la calidad del software, también se hace evidente la necesidad de contar con herramientas de software que permitan automatizar la evaluación y el seguimiento de estándares, ya sea para la definición de código fuente, trabajo en equipo o seguimiento de la metodología de trabajo, para esto se cuentan con diversos software que trataremos en este artículo como por ejemplo; SoapUI, Apache JMeter, HP LoadRunner, Sonarqube, Kiuwan, Redmine. Finalmente abordaremos a la calidad en el ámbito de las aplicaciones web y aplicaciones para la educación superior.

**Palabras Clave:** Ingeniería, software, calidad, requisitos, APIs, servicios web, REST, QA.

## ABSTRACT

This document presents a contextualized software quality definition, that is, depending on the area, it can be seen from the perspective of the product, process or software team. This definition is accompanied by definitions of features that can not be missing, such as: functionality, reliability, usability, efficiency, ease of maintenance and portability of the software product. It is by seeing the quality of the software from these perspectives that the understanding of standards that provide metrics for a more realistic evaluation is valued depending on what is being evaluated, whether the software as a product, the development team or the development process, For this reason, standards such as: ISO 12207 are described; to evaluate software life cycles, ISO 9001; to ensure quality in the design and development, ISO 9002; Quality systems- Quality assurance model in production and installation, ISO 9126; Product quality, ISO 14598; Evaluation of software products. As we have standards to address these perspectives of software quality, it is also evident the need to have software tools that automate the evaluation and monitoring of standards, whether for the definition of source code, teamwork or follow-up of the work methodology, for this there are several software that we will deal with in this article, for example; SoapUI, Apache JMeter, HP LoadRunner, Sonarqube, Kiuwan, Redmine. Finally we will address quality in the field of web applications and applications for higher education

**KEYWORDS:** Engineering, software, quality, requirements, APIs, web services, REST,QA.

## INTRODUCCIÓN

A pesar del avance tecnológico y diferentes herramientas y estándares hoy en día se siguen presentando dificultades para obtener productos de calidad a través de esta disciplina de la ingeniería de software razón por la que en esta artículo se pretende presentar y aclarar estos términos y conceptos iniciando con una conceptualización de la calidad en general para

luego ver su concepto en el contexto de la ingeniería del software, para luego verlo en más detalles desde las diferentes entidades involucradas ya sea la calidad como producto de software, la calidad en las metodologías y proceso de desarrollo o calidad en el equipo de desarrollo. Luego se valorará los diferentes estándares que existen para obtener productos de calidad y hacer la evaluación de la calidad que existe en la gestión del equipo o uso de procesos de desarrollo, también se verán herramientas de software que permitan automatizar el proceso de evaluación. Finalmente se describirá la calidad en las aplicaciones web y de educación superior como perspectivas y líneas de futuros trabajos en la calidad del software.

## **CONCEPTO DE CALIDAD**

Según la definición de la RAE (Diccionario de la Real Academia de la Lengua, 2018), define la calidad, como: “Propiedad o conjunto de propiedades inherentes a algo, que permiten juzgar su valor”. Esta definición está orientada al mercado.

### *A. Calidad de software*

La definición de la calidad del software según (IEEE STD 610-1990, 1990), es “el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario”

“La calidad de un sistema, aplicación o producto sólo es tan buena como los requerimientos que describen el problema, el diseño que modela la solución, el código que conduce a un programa ejecutable y las pruebas que ejercitan el software para descubrir errores” (Pressman, 2010).

Podemos ampliar que el concepto de calidad de software es para muchos autores algo muy subjetivos y poco tangible por no poder medir ni visualizar los resultados en corto tiempo de la implementación de un proyecto de software, es por lo que “La calidad del software no sólo se ve. Es el resultado de la buena administración del proyecto y de una correcta práctica de la ingeniería de software. La administración y práctica se aplican en el contexto de cuatro actividades principales que ayudan al equipo de software a lograr

una alta calidad en éste: métodos de la ingeniería de software, técnicas de administración de proyectos, acciones de control de calidad y aseguramiento de la calidad del software.” según dice (Pressman, 2010).

## CARACTERÍSTICA DE LA CALIDAD DE SOFTWARE

De acuerdo con (Blog Productora Digital, Software Factory (4r Soluciones), 2013) se refiere al estándar ISO/IEC 9126 presenta la calidad del software como un conjunto de seis características globales:

- **Funcionalidad.** Las funciones del software son aquellas que buscan satisfacer las necesidades del usuario.
- **Confiabilidad.** La capacidad del software de mantener su rendimiento bajo ciertas condiciones durante cierto período de tiempo.
- **Usabilidad.** Basada en el esfuerzo necesario para utilizar el *software* por parte de un grupo de usuarios.
- **Eficiencia.** Basada en la relación entre el nivel de rendimiento del *software* y el volumen de recursos utilizado, bajo ciertas condiciones.
- **Capacidad de mantenimiento.** Basada en el esfuerzo necesario para realizar modificaciones específicas.
- **Portabilidad.** Basada en la capacidad del software para ser transferido de un entorno a otro.

## COMPONENTES EN LA CALIDAD DE SOFTWARE

El desarrollo de software basado en componentes se ha convertido actualmente en uno de los mecanismos más efectivos para la construcción de grandes sistemas y aplicaciones de software. Una vez que la mayor parte de los aspectos funcionales de esta disciplina comienzan a estar bien definidos, la atención de la comunidad científica comienza a centrarse en los aspectos extra funcionales y de calidad, como un paso hacia una verdadera ingeniería según Bertoa (Bertoa, 2011)

## **CALIDAD DE PROCESO VS CALIDAD DE PRODUCTO**

No es lo mismo calidad del PRODUCTO software, que calidad del PROCESO software, que calidad del EQUIPO.

En el mundo del software, y en otras disciplinas también, cuando nos referirnos al amplio concepto de “calidad software” hay que ser muy conscientes de que en realidad es abstracto el concepto de calidad y se subdivide, principalmente, en tres tipos de calidad: la del proceso, la del producto y la de las personas/equipos.

Este tema, aun con muchos años de desarrollo, en el mundo empresarial no está tan claro y cuando se entrega el software, aquello que se especifica en los requisitos y parece tan claro resulta que produce enormes pérdidas de esfuerzo. Encontrando casos muy escasos de empresas a nivel latinoamericano que cumplen con certificaciones de nivel 6 según (Gonzales Meneses, León Padilla, Hernández Mora, & Medina Barrera, 2013) llamado “Garantía de la calidad del proceso y del producto” (CMMI Institute LLC., 2019) en el cual la empresa provee personal y gestión de comprensión objetiva del proceso y de los productos asociados (Ian, 2005) ocasionando que haya productos con mala calidad desde la percepción del usuario finales.

Sin pretender, ni poder, resolver todo, se hace un breve resumen de estas tres diferentes y determinantes áreas de la calidad software.

### *A. La calidad del proceso*

La calidad vista desde el mundo de los procesos nos dice que la calidad del producto software está determinada por la calidad del proceso (Parra, 2014). Por proceso se entienden las actividades, tareas, entrada, salida, procedimientos, etc., para desarrollar y mantener software.

Modelos, normas y metodologías típicas aquí son CMMI, ISO 15504 / ISO 12207, el ciclo de vida usado; incluso las metodologías ágiles entran en este apartado.

### *B. La calidad del producto*

Existen modelos de calidad de producto, destacando entre ellos la ISO 9126 (ISO, 2001), o la nueva serie ISO 25000 (ISO, 2005a), que especifica

diferentes dimensiones de la calidad de producto. Aunque aquí la dura tarea de evaluación recae en el uso de métricas software.

### *C. La calidad del equipo y/o personas*

Si hubiese que elegir, entre las claves que determinan el éxito de un proyecto de software, me aventuraría a decir que este sería el papel que juegan “las personas”.

En mayor o menor medida, prácticamente todo aquel que ha estudiado el éxito o fracaso de un proyecto software ha destacado el papel que las personas, el equipo de desarrollo, juegan en el mismo. Concluyendo, en la mayoría de las ocasiones, con que las personas son el factor más determinante.

No hay que olvidar que las personas son las que hacen el software. Las herramientas ayudan, las técnicas también, los procesos, pero sobre todo esto está las personas. “Las personas son la clave del éxito, el equipo humano que es el componente no lineal de primer orden en el desarrollo software.

### *D. Calidad en desarrollo de software*

Según la IEEE 729-83, la calidad del software es: “Grado con el cual el cliente o usuario percibe que el software satisface sus expectativas”. La norma ISO 8402:1984, la define como “Conjunto de propiedades y de características de un producto o servicio, que le confieren aptitud para satisfacer necesidades explícitas o implícitas”. (Pressman, 2010) la define como: “Concordancia del software producido con los requerimientos explícitamente establecidos, con los estándares de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente, que desea el usuario” Hablar acerca de la calidad que un software debe de tener, es un tema bastante amplio y en ocasiones desconcertante.

El tamaño para los títulos de las tablas, figuras y notas al pie de página es de 8 puntos. Todas las figuras y tablas deben aparecer centradas en la columna (las figuras y tablas de gran tamaño podrán extenderse sobre ambas columnas). Evite ubicar las figuras y tablas en medio de las columnas, siendo preferible su ubicación en la parte superior de la página. Se aconseja que inserte la figura o la tabla, junto con su descripción, en un cuadro de texto, tal como se hace en este documento.

## **ESTÁNDARES DE CALIDAD DE SOFTWARE**

Los estándares de calidad de software hacen parte de la ingeniería de software, utilización de estándares y metodologías para el diseño, programación, prueba y análisis del software desarrollado, con el objetivo de ofrecer una mayor confiabilidad, mantenibilidad en concordancia con los requisitos exigidos, con esto se eleva la productividad y el control en la calidad de software, parte de la gestión de la calidad se establecen a mejorar su eficacia y eficiencia.

Los estándares definen un conjunto de criterios que guían la forma en que se aplican procedimientos y metodologías al software desarrollado, la certificación de calidad permite una valoración independiente de la organización, donde se demuestra la capacidad de desarrollar productos y servicios de calidad. (Lozano, 2013).

### *A. ISO 12207 – Modelos de Ciclos de Vida del Software.*

Estándar para los procesos de ciclo de vida del software de la organización, Este estándar se concibió para aquellos interesados en adquisición de software, así como desarrolladores y proveedores. El estándar indica una serie de procesos desde la recopilación de requisitos hasta la culminación del software.

El estándar comprende 17 procesos lo cuales son agrupados en tres categorías:

- Principales
- De apoyo
- De organización

*B. ISO 9001: Sistema de calidad- Modelo de aseguramiento de la calidad en diseño/ desarrollo producción, instalación y servicio.*

Norma que se aplica a los sistemas de calidad, el cual se centra en todos los elementos de administración de calidad con los que una empresa debe contar, para tener un sistema efectivo que permita administrar y mejorar la calidad de sus productos y servicios.

La norma ISO 9001:2008 se basa en ocho principios de gestión de calidad:

- Enfoque al cliente
- Liderazgo
- Participación del personal
- Enfoque basado en procesos
- Enfoque de sistema para la gestión
- Mejora continua
- Enfoque basado en hechos para la toma de decisión
- Relaciones mutuamente beneficiosas con el proveedor.

*C. ISO 9002: Sistemas de calidad- Modelo de aseguramiento de la calidad en la producción e instalación.*

La norma ISO 9002 evalúa todos los recursos humanos y técnicos y los procedimientos administrativos, que garantizan la operación continua y efectiva del sistema de calidad en una empresa. Es necesaria la supervisión constante y el cumplimiento de especificaciones, las que servirán para documentar los procedimientos y aseguramiento que satisfacen los requisitos específicos del cliente. En el año 2000, ISO 9002 se unió a la norma ISO 9001, ya que eran similares. La norma ISO 9001 de aseguramiento de calidad estándar cubre tanto la producción, fabricación y partes de servicio de 9002, además de cubrir procesos de reacción del diseño.

*D. ISO 9126 – Calidad del producto.*

El estándar ISO 9126 ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software evalúa los productos de software, esta norma nos indica las características de la calidad y los lineamientos para su uso. El estándar identifica 6 atributos clave de calidad:



- **Funcionalidad:** el grado en que el software satisface las necesidades indicadas por los siguientes sub-atributos: idoneidad, corrección, Inter operatividad, conformidad y seguridad.
- **Confiabilidad:** cantidad de tiempo que el software está disponible para su uso. Está referido por los siguientes sub-atributos: madurez, tolerancia a fallos y facilidad de recuperación.
- **Usabilidad:** grado en que el software es fácil de usar. Viene reflejado por los siguientes sub-atributos: facilidad de comprensión, facilidad de aprendizaje y operatividad.
- **Eficiencia:** grado en que el software hace óptimo el uso de los recursos del sistema. Está indicado por los siguientes sub-atributos: tiempo de uso y recursos utilizados.
- **Facilidad de mantenimiento:** la facilidad con que una modificación puede ser realizada. Está indicada por los siguientes sub-atributos: facilidad de análisis, facilidad de cambio, estabilidad y facilidad de prueba.
- **Portabilidad:** la facilidad con que el software puede ser llevado de un entorno a otro. Está referido por los siguientes sub-atributos: facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio.

#### *E. ISO 14598 – Evaluación de productos de software*

La norma ISO/IEC 14598 es un estándar que proporciona un marco de trabajo para evaluar la calidad de todo tipo de producto software e indica los requisitos para los métodos de medición y el proceso de evaluación, proporcionando métricas y requisitos para los procesos de evaluación, a través de 6 etapas.

## **SOFTWARE PARA EVALUAR CALIDAD DE SOFTWARE**

Según Hernández (Hernández, 2018) podemos mencionar:

### *A. SoapUI*

Pertenece a las herramientas que nacieron para probar software. Está desarrollada en Java, y se utiliza para pruebas funcionales de APIs y servicios web (Soapui, 2019). SoapUI tiene una versión gratuita de código abierto, y una versión de pago con algunas funcionalidades que hacen que

sea mucho más productiva. Se trata de una opción absolutamente madura, cuya primera versión es de septiembre de 2005. Prácticamente imprescindible para los expertos en pruebas sobre APIs.

SoapUI tiene funcionalidades interesantes: Permite crear conjuntos de pruebas tan complicados como queramos, analizar la cobertura de tests sobre nuestro servicio SOAP o REST, cambiar el entorno de pruebas de forma rápidamente, crear mocks a partir de un WSDL o incluso facilitar ciertas pruebas de seguridad.

La alternativa a SoapUI es Postman, mucho más popular entre los desarrolladores que SoapUI. Postman nos permite construir y gestionar de una forma cómoda nuestras peticiones a servicios API REST.

#### *B. Apache JMeter - HP LoadRunner*

Apache JMeter (Apache, 2019) y HP LoadRunner (Microfocus, 2019) son 2 de los mejores programas para realizar pruebas de rendimiento y stress. JMeter es de código abierto y se puede descargar gratuitamente. Se utiliza para generar un gran volumen de carga que nos permita analizar y medir el rendimiento de aplicaciones web.

Load Runner es la alternativa para pruebas de rendimiento de HP. Existe la opción de utilizar LoadRunner en versión SaaS, de forma gratuita con su Community Edition, y ver de esta forma si es la herramienta adecuada para nosotros, sin ningún coste.

Una tercera opción para pruebas de rendimiento, también como SaaS es Octoperf. Basándose en JMeter, han creado una herramienta que no necesita de ninguna instalación y que nos permite crear escenarios, monitorizar nuestros entornos, ejecutar pruebas y analizar los resultados desde un mismo punto.

#### *C. Sonarqube - Kiuwan*

Sonarqube (Sonarqube, 2019) es una de las utilidades más populares para realizar análisis estático de código. Es open source, por lo que en principio es gratuito. Eso sí, tendremos que instalarlo en una máquina, y mantenerlo actualizado. Además, determinados plugins son de pago, como por ejemplo el plugin para analizar código Swift.

Kiuwan (Kiuwan, 2019) es otro analizador estático de código, pero en este caso se trata de un servicio que podemos usar para analizar nuestro código

sin preocuparnos por instalaciones ni actualizaciones. Podemos subir nuestro código a la nube para analizarlo, o descargar una aplicación que analizará nuestro código localmente y subirá los resultados a Kiuwan.

Tanto Sonarqube (sonarlint) como Kiuwan tienen integración con distintos IDEs, que nos permiten detectar incidencias en nuestro código mientras lo escribimos, sin tener que esperar a análisis posteriores. También en ambos casos existen plugins para herramientas de integración continua como Jenkins.

#### *D. Redmine*

Está más enfocada en equipos de QA. Es un software para la gestión de proyectos que incluye un sistema de seguimiento de errores. Redmine destaca por ser software libre y de código abierto, de alta capacidad para integrarse con cualquier proyecto (Redmine, 2019).

## **CALIDAD EN EL DESARROLLO DE APLICACIONES PARA WEB**

Con el transcurrir del tiempo los equipos de desarrollos de las compañías que construyen software para los diferentes ámbitos de los negocios o sectores de la economía han realizado cambios de acuerdo con las innovaciones tecnológicas que exigen publicar dichos productos en ambientes que sean accedidos por diferentes tipos de usuarios en cualquier parte del mundo, siempre y cuando tenga las credenciales para la misma. Es por eso que existen diferentes tipos de tecnología que apoyan y complementan este tipo de desarrollo en la industria del software.

La calidad en aplicaciones Web los avances en Internet han conducido a un desarrollo impactante de sistemas y aplicaciones basadas en la Web, suceso que se presenta como el más significativo en la historia de la computación.

Muchas de las nuevas tecnologías y estándares de la Web han surgido en los últimos años para mejorar el apoyo a nuevas aplicaciones Web: XML, servicios Web, Web semántica, técnicas de personalización de la Web,

minería Web, inteligencia, contextaware y móviles y servicios Web. Las aplicaciones web son diferentes de otras categorías de software; son eminentemente de red, las gobiernan los datos y se encuentran en evolución continua. El estudio de la calidad de productos y procesos de desarrollo para la Web es muy reciente y todavía no se dispone de métodos de evaluación ampliamente difundidos para este tipo de entorno, por lo tanto, existe la necesidad de metodologías efectivas para la obtención de aplicaciones Web de calidad.

La Ingeniería Web surge debido a la necesidad de lograr enfoques disciplinados y nuevos métodos y herramientas para desarrollar, desplegar y evaluar los sistemas y aplicaciones basados en la Web.

## **CALIDAD DE SOFTWARE EN LAS INSTITUCIONES EDUCATIVAS SUPERIOR**

En el artículo de Marques (Marques, 2002) se utilizan las expresiones software educativo, programas educativos y programas didácticos como sinónimos para designar genéricamente los programas para ordenador creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje. Esta definición engloba todos los programas que han estado elaborados con fin didáctico, desde los tradicionales programas basados en los modelos conductistas de la enseñanza, los programas de Enseñanza Asistida por Ordenador (EAO), hasta los programas experimentales de Enseñanza Inteligente Asistida por Ordenador (EIAO), que, utilizando técnicas propias del campo de los Sistemas Expertos y de la Inteligencia Artificial en general, pretenden imitar la labor tutorial personalizada que realizan los profesores y presentan modelos de representación del conocimiento en consonancia con los procesos cognitivos que desarrollan los alumnos. No obstante, según esta definición, más basada en un criterio de finalidad que de funcionalidad, se excluyen del software educativo todos los programas de uso general en el mundo empresarial que también se utilizan en los centros educativos con funciones didácticas o instrumentales como, por ejemplo: procesadores de

textos, gestores de bases de datos, hojas de cálculo, editores gráficos... Estos programas, aunque puedan desarrollar una función didáctica, no han estado elaborados específicamente con esta finalidad.

## **CONCLUSIONES**

Actualmente en el ámbito de la informática uno de los principales retos que se enfrenta en el desarrollo de software es la calidad, la cual es mal entendida y por tanto afrontada. La falta de calidad es algo que afecta a todos los que están involucrados en el proyecto y tiene sus propias características que deberían integrar las diferentes áreas que la afectan, ya que se pudiera hablar de calidad en el proceso, calidad en el producto o en el equipo de desarrollo, todo esto para que finalmente las consecuencias y pérdidas no sean críticas y el nivel de satisfacción y logro de objetivos en los proyectos sean los esperados, pudiendo satisfacer las necesidades del usuario final y enriquecer con mejores prácticas en el proceso de desarrollo que se utiliza por parte del equipo. Para tener una calidad de software es necesario definir parámetro o criterios de medición. Debido a la creciente industria del software la cual ha evolucionado a través de los años y necesita de estándares de calidad de software con los cuales se pueda definir de forma clara los patrones, normas, modelos que se han de aplicar para el bien de todas las partes. Con la implementación de modelos de calidad de software se permitirá que las empresas de software adquieran un importante desarrollo de competitividad y eficiencia.

## REFERENCIAS BIBLIOGRÁFICAS

- Apache. (2019). Apache jmeter. Recuperado el 25 de 06 de 2019, de <https://jmeter.apache.org/>
- Bertoa, M. F. (6 de Octubre de 2011). Aspectos de Calidad en el Desarrollo de Software Basado en Componentes. Obtenido de [https://www.ecured.cu/Calidad\\_en\\_el\\_Desarrollo\\_Software\\_Basado\\_en\\_Componentes](https://www.ecured.cu/Calidad_en_el_Desarrollo_Software_Basado_en_Componentes)
- Blog Productora Digital, Software Factory (4r Soluciones). (22 de mayo de 2013). Blog Productora Digital, Software Factory. (4. r. Soluciones, Productor) Recuperado el 2019 de 06 de 28, de <http://www.4rsoluciones.com/blog/como-medir-la-calidad-en-software-2/>
- CMMI Institute LLC. (2019). CMMI institute. Obtenido de <https://cmmiinstitute.com/>
- Diccionario de la Real Academia de la Lengua. (2018). Obtenido de Real academia de la Lengua Española: <https://dle.rae.es/?id=6nVpk8P|6nXVL1Z>
- Gonzales Meneses, Y., León Padilla, N., Hernández Mora, J., & Medina Barrera, M. (2013). Ranking mundial en certificaciones CMMI-DEV. Revista Iberoamericana de Producción Académica y Gestión Educativa. Obtenido de <https://www.pag.org.mx/index.php/PAG/article/download/123/170>
- Hernández, R. (2018). 12 herramientas imprescindibles para asegurar la calidad del software (y sus alternativas). Obtenido de <https://www.genbeta.com/desarrollo/12-herramientas-imprescindibles-para-asegurar-la-calidad-del-software-y-sus-alternativas>
- Ian, S. (2005). Ingeniería del software. Madrid, España: Pearson.
- Kiuwan. (2019). Kiuwan. Recuperado el 30 de 06 de 2019, de <https://www.kiuwan.com/>
- Lozano, L. A. (2013). ESTÁNDARES DE CALIDAD DEL SOFTWARE. Obtenido de <http://estandarescalidadsoftware.blogspot.com/>

- Marques, P. (2002). El software educativo. (U. A. Barcelona, Ed.)  
Obtenido de  
[http://www.dirinfo.unsl.edu.ar/profesorado/INfyEduc/teorias/clasif\\_software\\_educativo\\_de\\_pere.pdf](http://www.dirinfo.unsl.edu.ar/profesorado/INfyEduc/teorias/clasif_software_educativo_de_pere.pdf)
- Microfocus. (2019). Loadrunner load testing. Recuperado el 29 de 06 de 2019, de <https://www.microfocus.com/en-us/products/loadrunner-load-testing/download>
- Parra, F. S. (2014). I Jornadas sobre Calidad del Producto Software e ISO 25000. Santiago de Compostela: 233 Grados de TI S.L.
- Pressman, R. S. (2010). Ingeniería del software un Enfoque Práctico (Septima ed.). Mexico D.F: M Graw Hill Educacion.
- Redmine. (2019). Redmine. Recuperado el 28 de 06 de 2019, de <http://www.redmine.org/>
- Soapui. (2019). Soapui. Recuperado el 01 de 06 de 2019, de <https://www.soapui.org>
- Sonarqube. (2019). Sonarqube. Recuperado el 30 de 06 de 2019, de [www.sonarqube.org/](http://www.sonarqube.org/)